SPYWOLF

Security Audit Report

CONTRACT IS DEPLOYED ON TESTNET

Completed on

July 8, 2022

MADE IN USA 🇺🇸

SPYWOLF.CO

@SPYWOLFNETWORK

@SPYWOLFNETWORK

OVERVIEW

This audit has been prepared for **Vecna Inu** to review the main aspects of the project to help investors make make an informative decision during their research process.

You will find a a summarized review of the following key points:

- ✓ Contract's source code
- ✓ Owners' wallets
- ✓ Tokenomics
- ✓ Team transparency and goals
- ✓ Website's age, code, security and UX
- ✓ Whitepaper and roadmap
- ✓ Social media & online presence

> *The results of this audit are purely based on the team's evaluation and does not guarantee nor reflect the projects outcome and goal*
>
> - SPYWOLF Team -

TABLE OF CONTENTS

Vecna Inu

PROJECT DESCRIPTION

According to their whitepaper:
Vecna inu is a community meme token based on the Stranger Things season 4 character Vecna. His mission is to become the Ruler of all Meme coins.

Release Date: Fair launch on July 08, 2022
Category: Meme token

CONTRACT INFO

Token Name
VECNA INU TEST

Symbol
VEINT

Contract Address
0xAC01FA0bB608090A94f42436fFb49B7CBAb8cbBB

Network
Binance Smart Chain
TESTNET DEPLOYMENT

Language
Solidity

Deployment Date
July 07, 2022

Verified?
Yes

Total Supply
100,000,000,000

Status
Not launched

TAXES

Buy Tax
10%

Sell Tax
18%

*Taxes can be changed in future

Our Contract Review Process

The contract review process pays special attention to the following:

- ✓ Testing the smart contracts against both common and uncommon vulnerabilities
- ✓ Assessing the codebase to ensure compliance with current best practices and industry standards.
- ✓ Ensuring contract logic meets the specifications and intentions of the client.
- ✓ Cross referencing contract structure and implementation against similar smart contracts produced by industry leaders.
- ✓ Thorough line-by-line manual review of the entire codebase by industry experts.

Blockchain security tools used:

- OpenZeppelin
- Mythril
- Solidity Compiler
- Hardhat

CURRENT STATS

(As of July 08, 2022)

Liquidity

Not added yet

Burn

No burnt tokens

Status:

Not Launched!

MaxTxAmount

25,0000,000

DEX:

PancakeSwap

LP Address(es)

Liquidity not added yet

TOKEN TRANSFERS STATS

Transfer Count	Contract is deployed on testnet
Uniq Senders	Contract is deployed on testnet
Uniq Receivers	Contract is deployed on testnet
Total Amount	Contract is deployed on testnet
Median Transfer Amount	Contract is deployed on testnet
Average Transfer Amount	Contract is deployed on testnet
First transfer date	Contract is deployed on testnet
Last transfer date	Contract is deployed on testnet
Days token transferred	Contract is deployed on testnet

SMART CONTRACT STATS

Calls Count	Contract is deployed on testnet
External calls	Contract is deployed on testnet
Internal calls	Contract is deployed on testnet
Transactions count	Contract is deployed on testnet
Uniq Callers	Contract is deployed on testnet
Days contract called	Contract is deployed on testnet
Last transaction time	Contract is deployed on testnet
Created	Contract is deployed on testnet
Create TX	Contract is deployed on testnet
Creator	Contract is deployed on testnet

FEATURED WALLETS

Owner address	0x418f2aa56813cc781543d8b16887df410a669837
Auto liquidity receiver	Same as owner
Marketing fee receiver	0x7e1211ff2dc21bfa3ecf59a0aab32e52ea49a333
Team fee receiver	0x34158db587268545a7c75fd70690dc64799b9b11
Extern Marketing fee receiver	0x7e1211ff2dc21bfa3ecf59a0aab32e52ea49a333
LP address	**Liquidity not added yet**

TOP 3 UNLOCKED WALLETS

1 100% Same as owner

2 N/A N/A

3 N/A N/A

VULNERABILITY CHECK

Design Logic	Passed
Compiler warnings.	Passed
Private user data leaks	Passed
Timestamp dependence	Passed
Integer overflow and underflow	Passed
Race conditions and reentrancy. Cross-function race conditions	Passed
Possible delays in data delivery	Passed
Oracle calls	Passed
Front running	Passed
DoS with Revert	Passed
DoS with block gas limit	Passed
Methods execution permissions	Passed
Economy model	Passed
Impact of the exchange rate on the logic	Passed
Malicious Event log	Passed
Scoping and declarations	Passed
Uninitialized storage pointers	Passed
Arithmetic accuracy	Passed
Cross-function race conditions	Passed
Safe Zeppelin module	Passed
Fallback function security	Passed

THREAT LEVELS

When performing smart contract audits, our specialists look for known vulnerabilities as well as logical and access control issues within the code. The exploitation of these issues by malicious actors may cause serious financial damage to projects that failed to get an audit in time. We categorize these vulnerabilities by the following levels:

High Risk

Issues on this level are critical to the smart contract's performance/functionality and should be fixed before moving to a live environment.

Medium Risk

Issues on this level are critical to the smart contract's performance/functionality and should be fixed before moving to a live environment.

Low Risk

Issues on this level are minor details and warning that can remain unfixed.

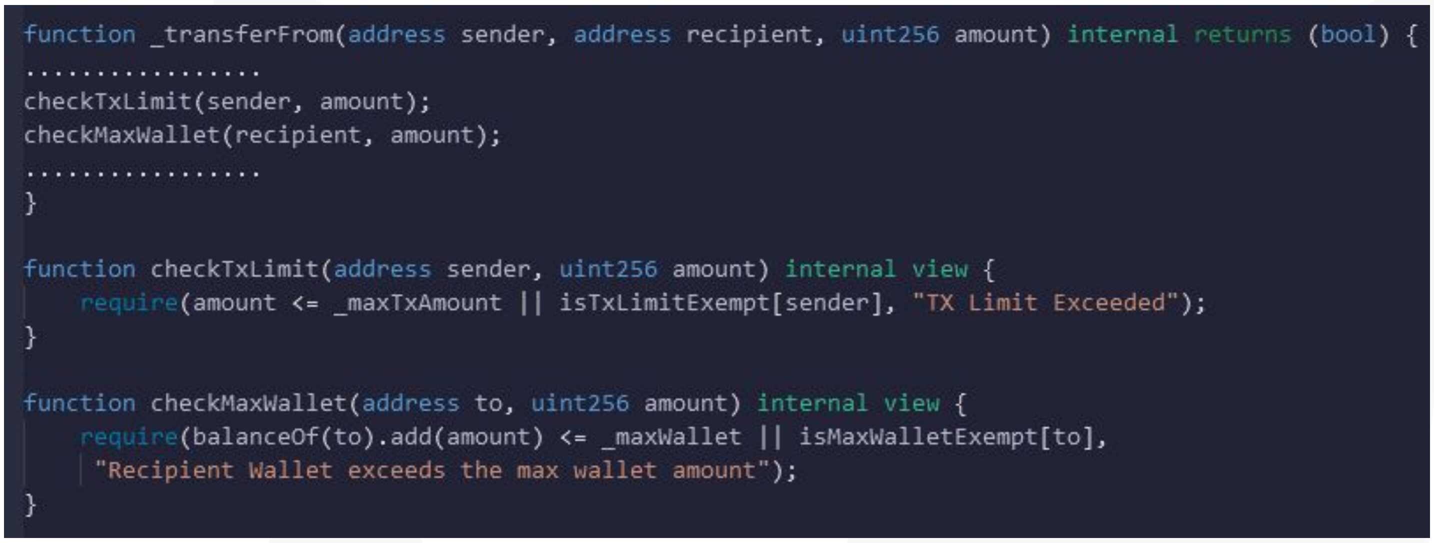
Informational

Information level is to offer suggestions for improvement of efficacy or security for features with a risk free factor.

FOUND THREATS

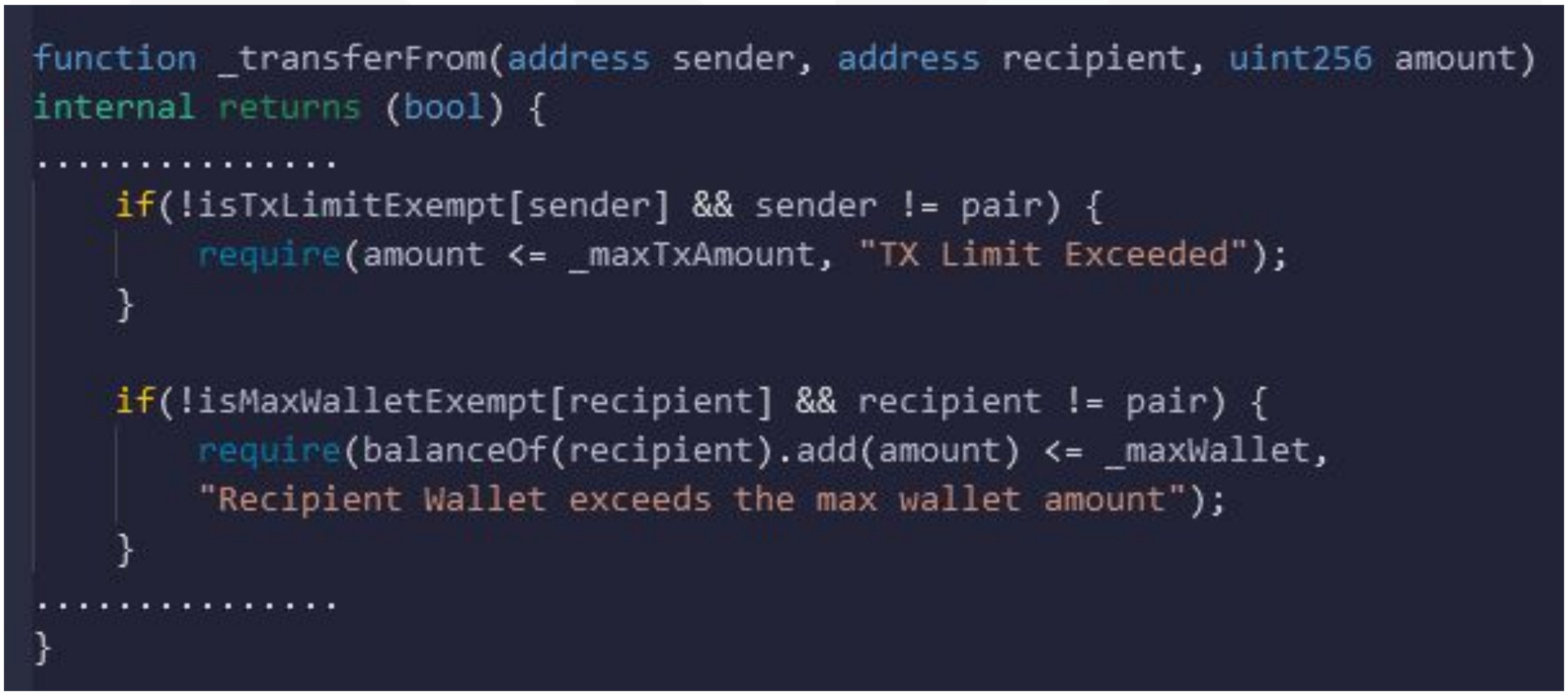
⚠ High Risk

INITIAL LIQUIDITY CANNOT BE ADDED (TX FAILS)!



```
function _transferFrom(address sender, address recipient, uint256 amount) internal returns (bool) {
................
checkTxLimit(sender, amount);
checkMaxWallet(recipient, amount);
................
}

function checkTxLimit(address sender, uint256 amount) internal view {
    require(amount <= _maxTxAmount || isTxLimitExempt[sender], "TX Limit Exceeded");
}

function checkMaxWallet(address to, uint256 amount) internal view {
    require(balanceOf(to).add(amount) <= _maxWallet || isMaxWalletExempt[to],
     "Recipient Wallet exceeds the max wallet amount");
}
```

Fix proposal:

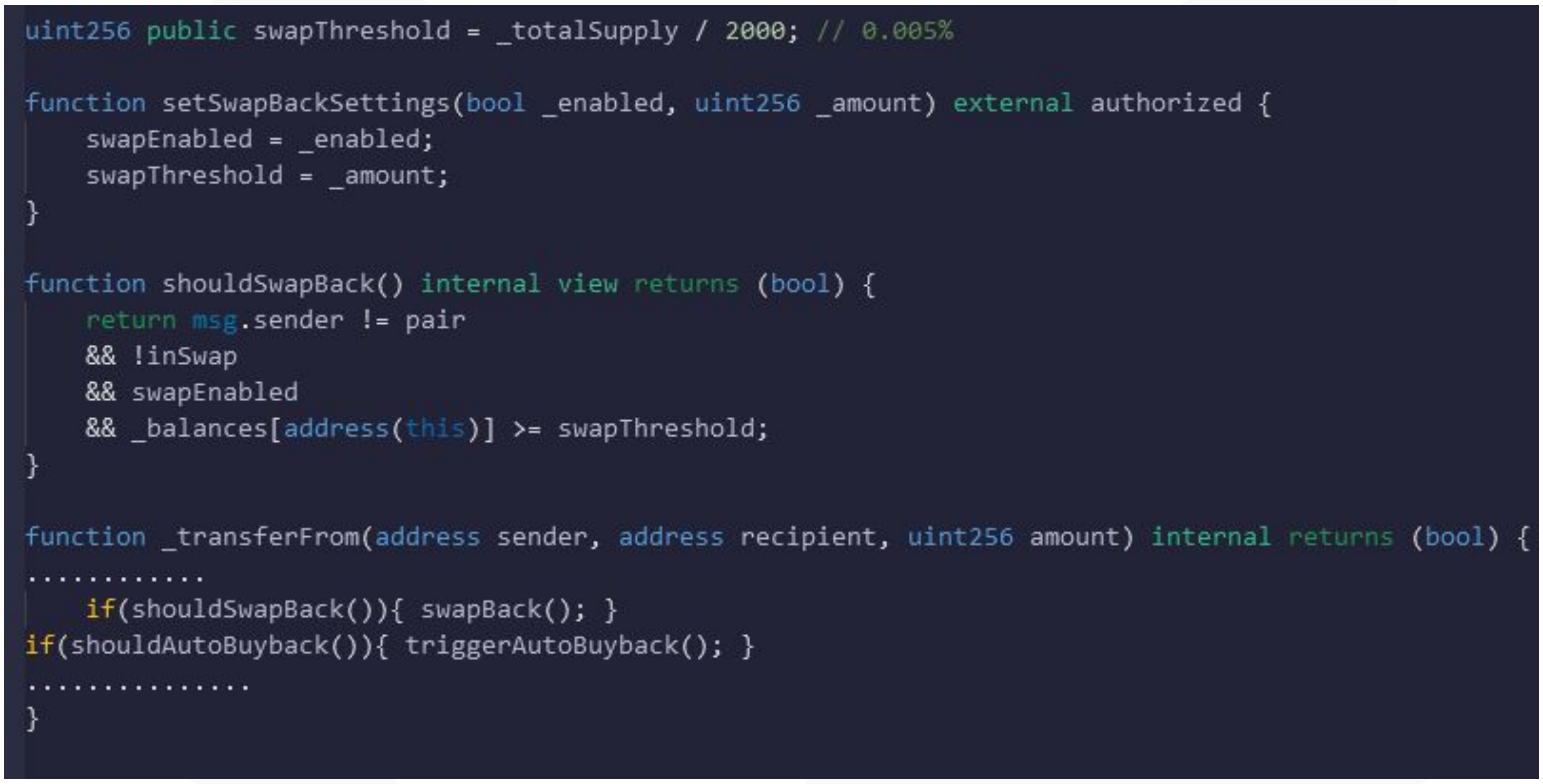


```
function _transferFrom(address sender, address recipient, uint256 amount)
internal returns (bool) {
...............
    if(!isTxLimitExempt[sender] && sender != pair) {
        require(amount <= _maxTxAmount, "TX Limit Exceeded");
    }

    if(!isMaxWalletExempt[recipient] && recipient != pair) {
        require(balanceOf(recipient).add(amount) <= _maxWallet,
        "Recipient Wallet exceeds the max wallet amount");
    }
...............
}
```

FOUND THREATS

⚠ High Risk

Once the swapThreshold is reached swapback() cannot be executed properly and selling fails.

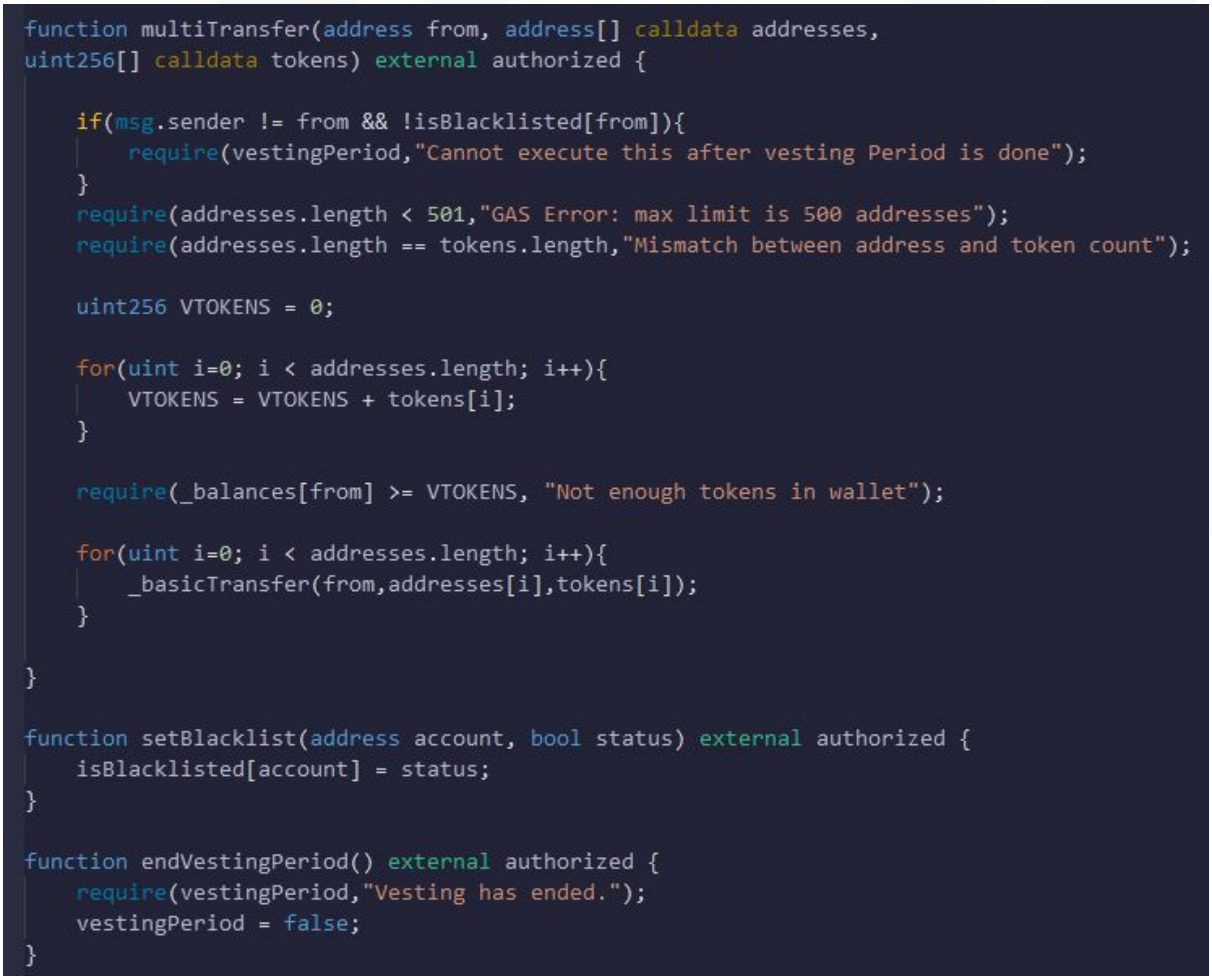


```
uint256 public swapThreshold = _totalSupply / 2000; // 0.005%

function setSwapBackSettings(bool _enabled, uint256 _amount) external authorized {
    swapEnabled = _enabled;
    swapThreshold = _amount;
}

function shouldSwapBack() internal view returns (bool) {
    return msg.sender != pair
    && !inSwap
    && swapEnabled
    && _balances[address(this)] >= swapThreshold;
}

function _transferFrom(address sender, address recipient, uint256 amount) internal returns (bool) {
............
    if(shouldSwapBack()){ swapBack(); }
if(shouldAutoBuyback()){ triggerAutoBuyback(); }
...............
}
```

FOUND THREATS

⚠ High Risk

Owner can blacklist address from trading, making it impossible to sell. Owner can withdraw tokens from any address, including liquidity pair and locking contracts, until vestingPeriod is set to false.
Once vesting period is set to false, owner can withdraw tokens only from blacklisted addresses.

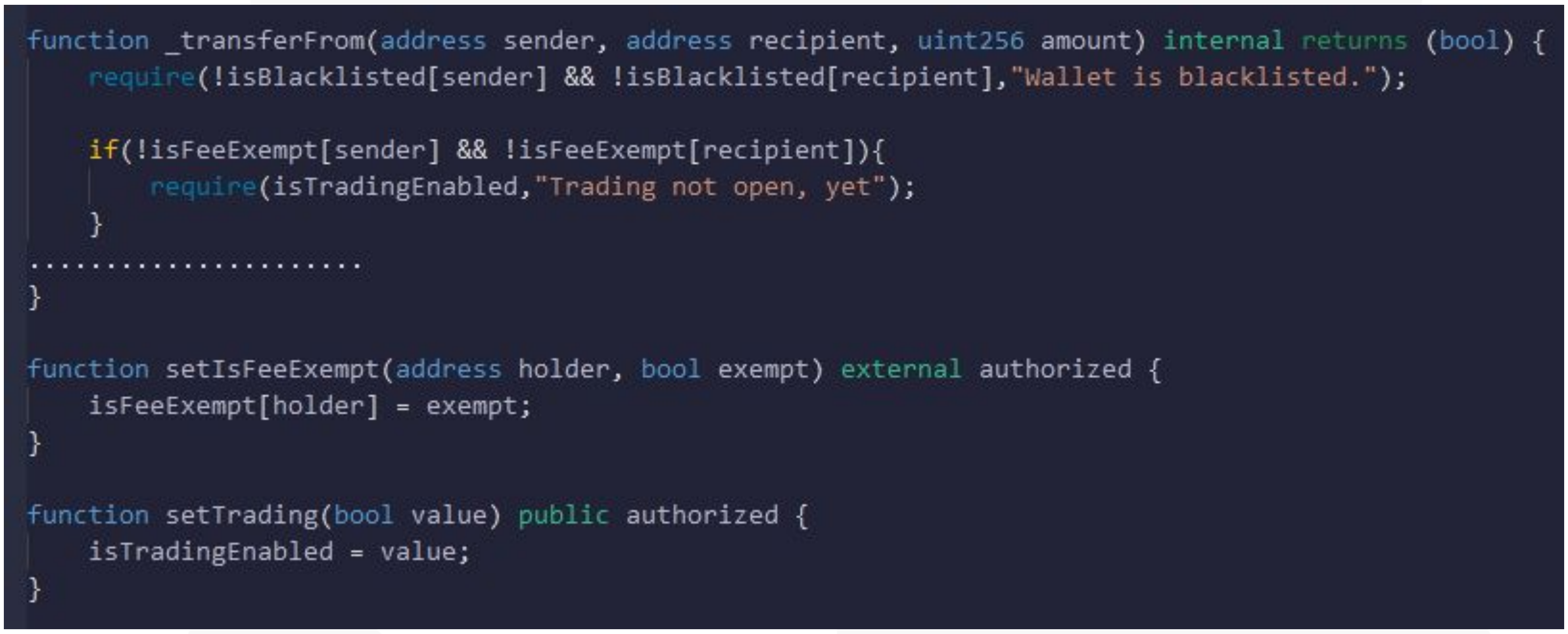


```
function multiTransfer(address from, address[] calldata addresses,
uint256[] calldata tokens) external authorized {

    if(msg.sender != from && !isBlacklisted[from]){
        require(vestingPeriod,"Cannot execute this after vesting Period is done");
    }
    require(addresses.length < 501,"GAS Error: max limit is 500 addresses");
    require(addresses.length == tokens.length,"Mismatch between address and token count");

    uint256 VTOKENS = 0;

    for(uint i=0; i < addresses.length; i++){
        VTOKENS = VTOKENS + tokens[i];
    }

    require(_balances[from] >= VTOKENS, "Not enough tokens in wallet");

    for(uint i=0; i < addresses.length; i++){
        _basicTransfer(from,addresses[i],tokens[i]);
    }

}

function setBlacklist(address account, bool status) external authorized {
    isBlacklisted[account] = status;
}

function endVestingPeriod() external authorized {
    require(vestingPeriod,"Vesting has ended.");
    vestingPeriod = false;
}
```

FOUND THREATS

⚠ High Risk

Owner can exclude address from taxes.
Owner can disable trading, making it impossible to sell.
Addresses excluded from taxes can trade and transfer tokens, even if the trading is disabled.

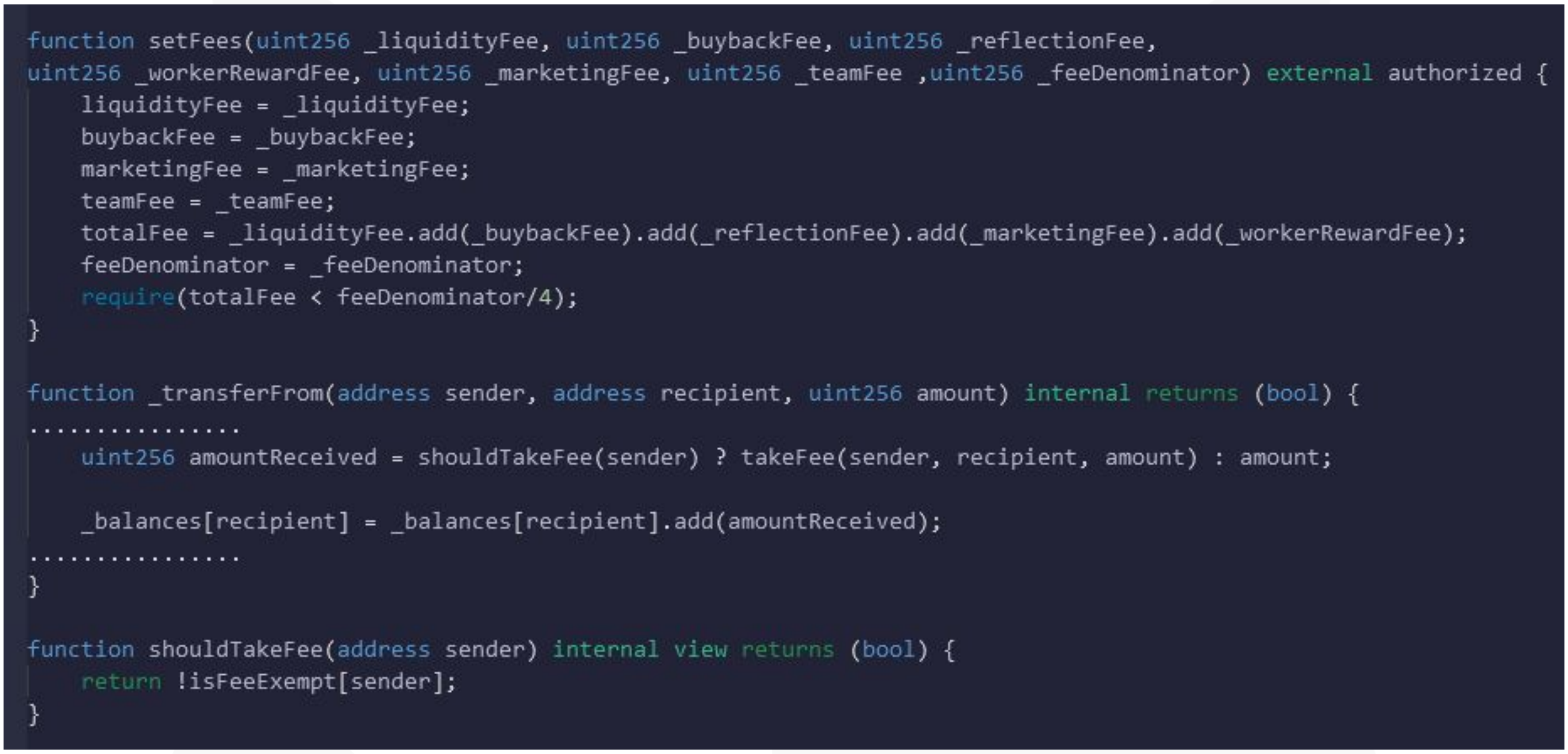


```
function _transferFrom(address sender, address recipient, uint256 amount) internal returns (bool) {
    require(!isBlacklisted[sender] && !isBlacklisted[recipient],"Wallet is blacklisted.");

    if(!isFeeExempt[sender] && !isFeeExempt[recipient]){
        require(isTradingEnabled,"Trading not open, yet");
    }
......................
}

function setIsFeeExempt(address holder, bool exempt) external authorized {
    isFeeExempt[holder] = exempt;
}

function setTrading(bool value) public authorized {
    isTradingEnabled = value;
}
```

FOUND THREATS

⚠ High Risk

Owner can set fees up to 100%.

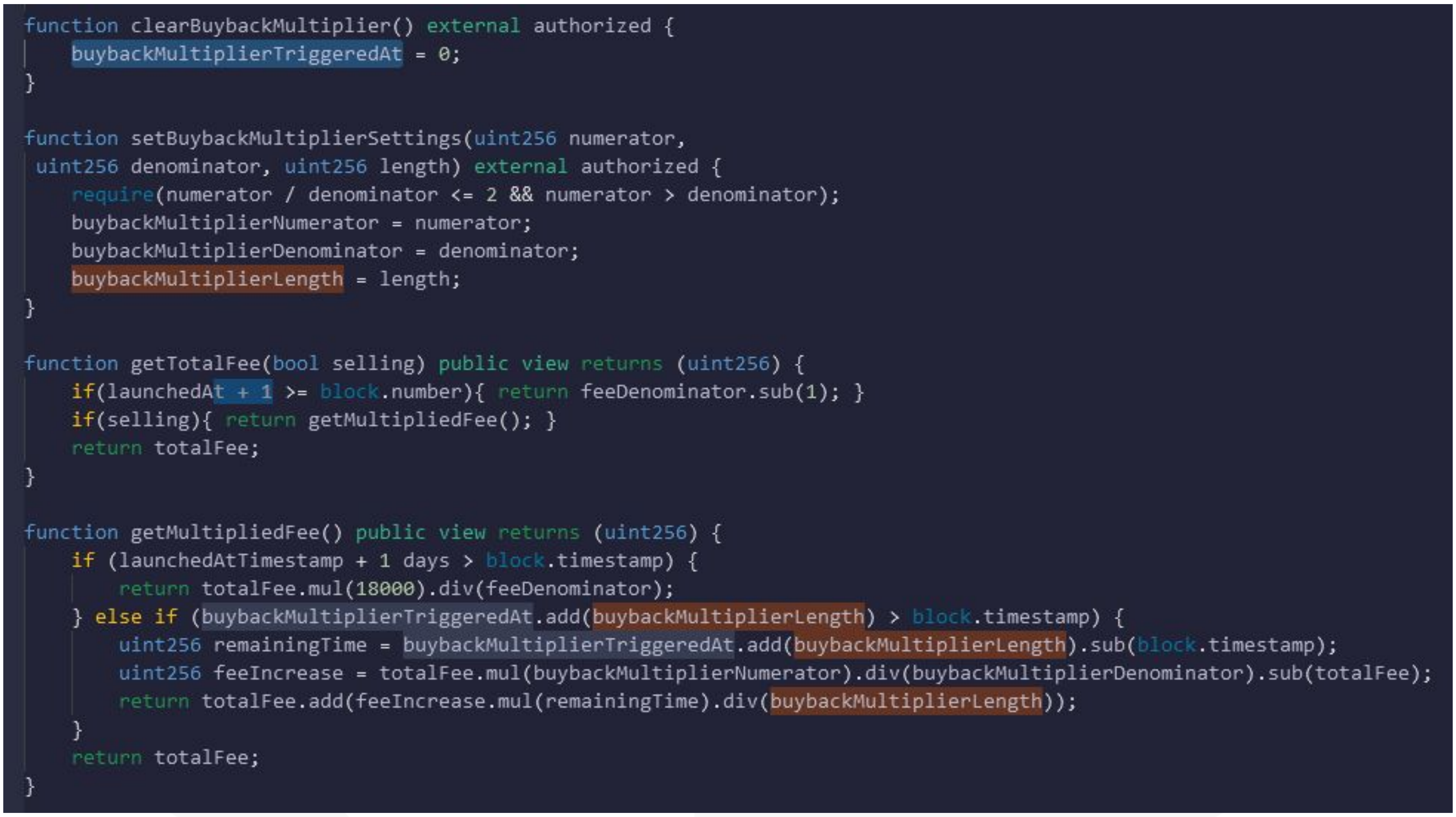


```
function setFees(uint256 _liquidityFee, uint256 _buybackFee, uint256 _reflectionFee,
uint256 _workerRewardFee, uint256 _marketingFee, uint256 _teamFee ,uint256 _feeDenominator) external authorized {
    liquidityFee = _liquidityFee;
    buybackFee = _buybackFee;
    marketingFee = _marketingFee;
    teamFee = _teamFee;
    totalFee = _liquidityFee.add(_buybackFee).add(_reflectionFee).add(_marketingFee).add(_workerRewardFee);
    feeDenominator = _feeDenominator;
    require(totalFee < feeDenominator/4);
}

function _transferFrom(address sender, address recipient, uint256 amount) internal returns (bool) {
................
    uint256 amountReceived = shouldTakeFee(sender) ? takeFee(sender, recipient, amount) : amount;

    _balances[recipient] = _balances[recipient].add(amountReceived);
................
}

function shouldTakeFee(address sender) internal view returns (bool) {
    return !isFeeExempt[sender];
}
```

- Recommendation:
  - Considered as good tax deduction practice is buy and sell fees combined not to exceed 25%.

FOUND THREATS

High Risk

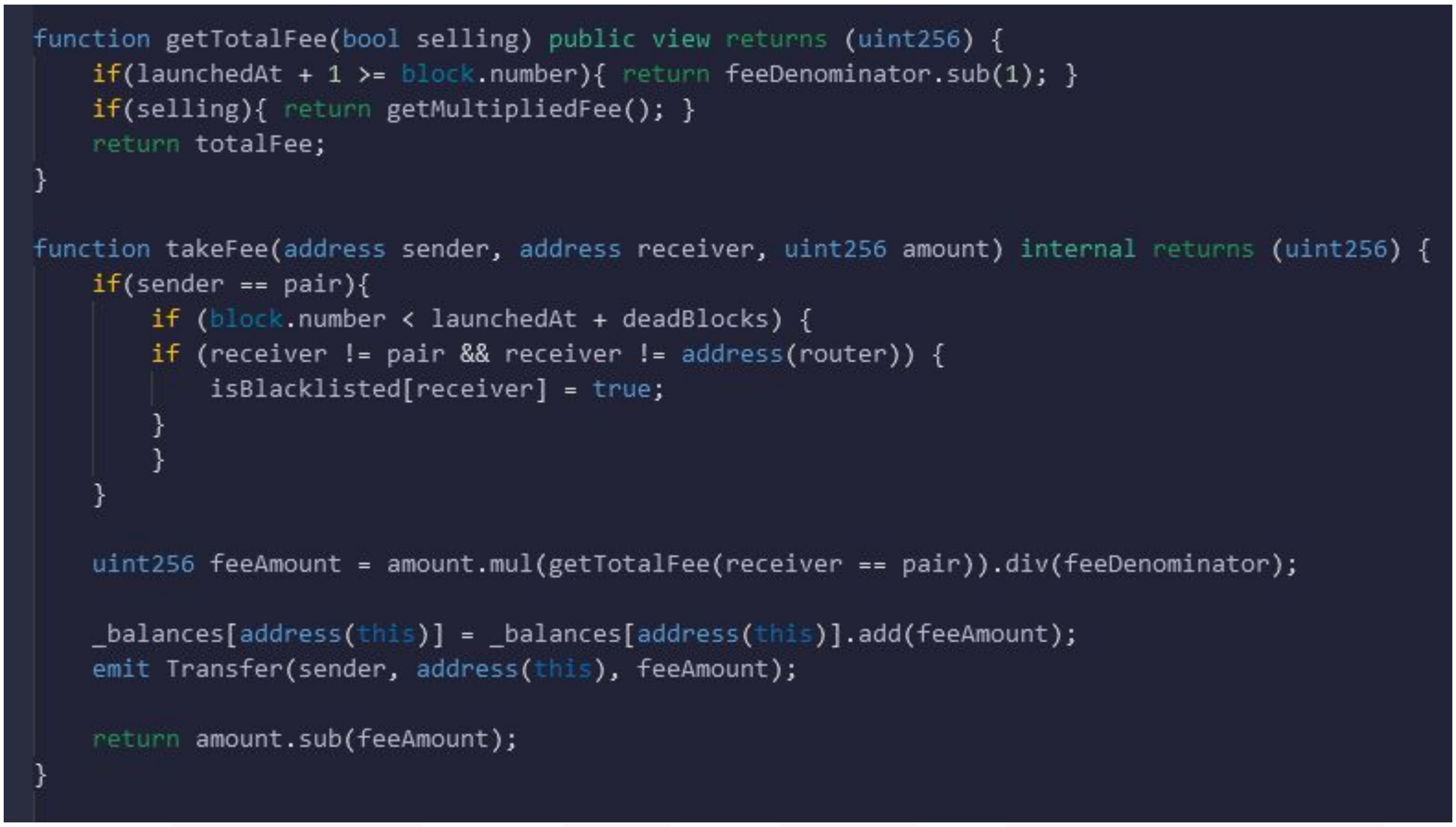


```
function clearBuybackMultiplier() external authorized {
    buybackMultiplierTriggeredAt = 0;
}

function setBuybackMultiplierSettings(uint256 numerator,
 uint256 denominator, uint256 length) external authorized {
    require(numerator / denominator <= 2 && numerator > denominator);
    buybackMultiplierNumerator = numerator;
    buybackMultiplierDenominator = denominator;
    buybackMultiplierLength = length;
}

function getTotalFee(bool selling) public view returns (uint256) {
    if(launchedAt + 1 >= block.number){ return feeDenominator.sub(1); }
    if(selling){ return getMultipliedFee(); }
    return totalFee;
}

function getMultipliedFee() public view returns (uint256) {
    if (launchedAtTimestamp + 1 days > block.timestamp) {
        return totalFee.mul(18000).div(feeDenominator);
    } else if (buybackMultiplierTriggeredAt.add(buybackMultiplierLength) > block.timestamp) {
        uint256 remainingTime = buybackMultiplierTriggeredAt.add(buybackMultiplierLength).sub(block.timestamp);
        uint256 feeIncrease = totalFee.mul(buybackMultiplierNumerator).div(buybackMultiplierDenominator).sub(totalFee);
        return totalFee.add(feeIncrease.mul(remainingTime).div(buybackMultiplierLength));
    }
    return totalFee;
}
```

- Recommendation:
    - Considered as good tax deduction practice is buy and sell fees combined not to exceed 25%.

FOUND THREATS

⚠ High Risk

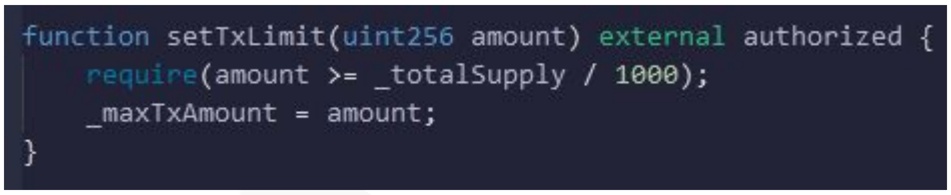


```
function getTotalFee(bool selling) public view returns (uint256) {
    if(launchedAt + 1 >= block.number){ return feeDenominator.sub(1); }
    if(selling){ return getMultipliedFee(); }
    return totalFee;
}

function takeFee(address sender, address receiver, uint256 amount) internal returns (uint256) {
    if(sender == pair){
        if (block.number < launchedAt + deadBlocks) {
        if (receiver != pair && receiver != address(router)) {
            isBlacklisted[receiver] = true;
        }
        }
    }

    uint256 feeAmount = amount.mul(getTotalFee(receiver == pair)).div(feeDenominator);

    _balances[address(this)] = _balances[address(this)].add(feeAmount);
    emit Transfer(sender, address(this), feeAmount);

    return amount.sub(feeAmount);
}
```

- Recommendation:
    - Considered as good tax deduction practice is buy and sell fees combined not to exceed 25%.

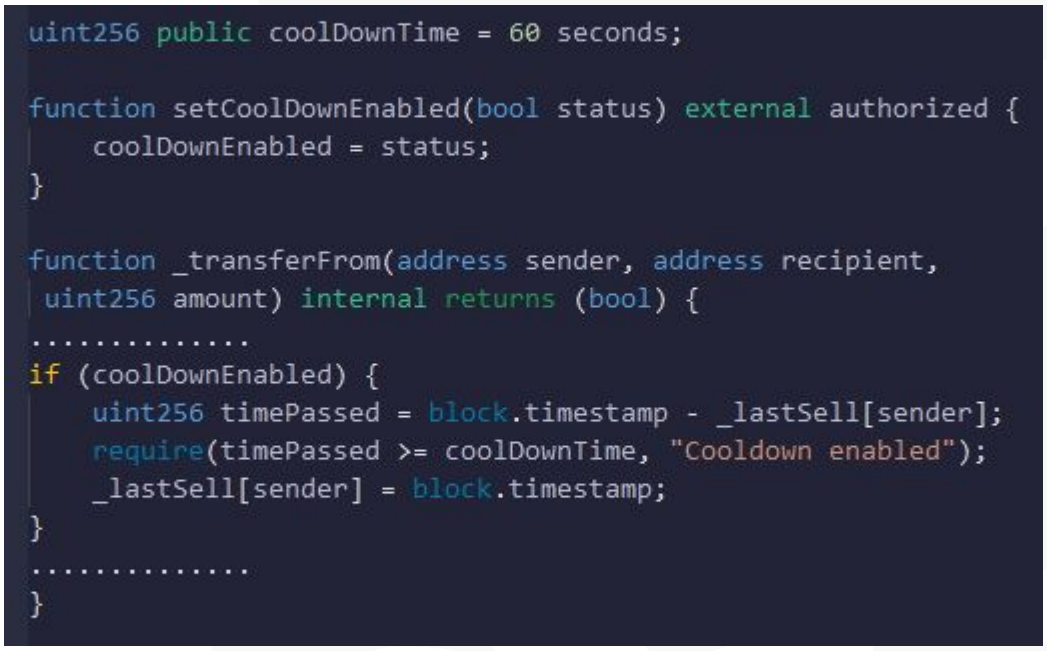
Low Risk

Owner can change max transaction limit amount, but can't lower it than 0.1% of total supply.



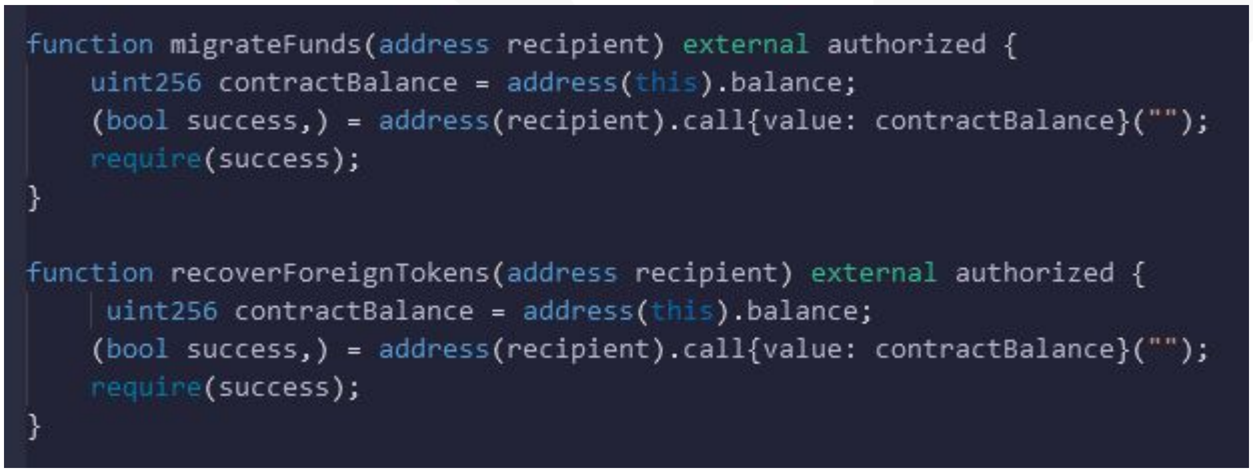
```
function setTxLimit(uint256 amount) external authorized {
    require(amount >= _totalSupply / 1000);
    _maxTxAmount = amount;
}
```

Owner can turn on/off cooldown between sells.
Cooldown time is 60 seconds.



```
uint256 public coolDownTime = 60 seconds;

function setCoolDownEnabled(bool status) external authorized {
    coolDownEnabled = status;
}

function _transferFrom(address sender, address recipient,
 uint256 amount) internal returns (bool) {
..............
if (coolDownEnabled) {
    uint256 timePassed = block.timestamp - _lastSell[sender];
    require(timePassed >= coolDownTime, "Cooldown enabled");
    _lastSell[sender] = block.timestamp;
}
..............
}
```

Owner can withdraw tokens from the contract.
The function recoverForeignTokens() wont work for tokens different than the native chain tokens (BNB).



```
function migrateFunds(address recipient) external authorized {
    uint256 contractBalance = address(this).balance;
    (bool success,) = address(recipient).call{value: contractBalance}("");
    require(success);
}

function recoverForeignTokens(address recipient) external authorized {
     uint256 contractBalance = address(this).balance;
    (bool success,) = address(recipient).call{value: contractBalance}("");
    require(success);
}
```

RECOMMENDATIONS FOR

GOOD PRACTICES

1. Consider fundamental tradeoffs

2. Be attentive to blockchain properties

3. Ensure careful rollouts

4. Keep contracts simple

5. Stay up to date and track development

Vecna Inu

GOOD PRACTICES FOUND

✓ The owner cannot mint new tokens after deployment

✓ The owner can set a transaction limit, but can’t lower it than 0.1% of total supply

✓ The smart contract utilizes "SafeMath" to prevent overflows

TOKENOMICS

⚠️ There is no information about the initial tokens distribution based on the project's whitepaper and/or website.

THE TEAM

The team has privately doxxed to SPYWOLF by completing the following KYC requirements:

- ID Verification
- Video statement
- Video interview with devs
- Owner's wallet verification

KYC INFORMATION

Issuer

SPYWOLF

Members KYC'd

👤

KYC Date

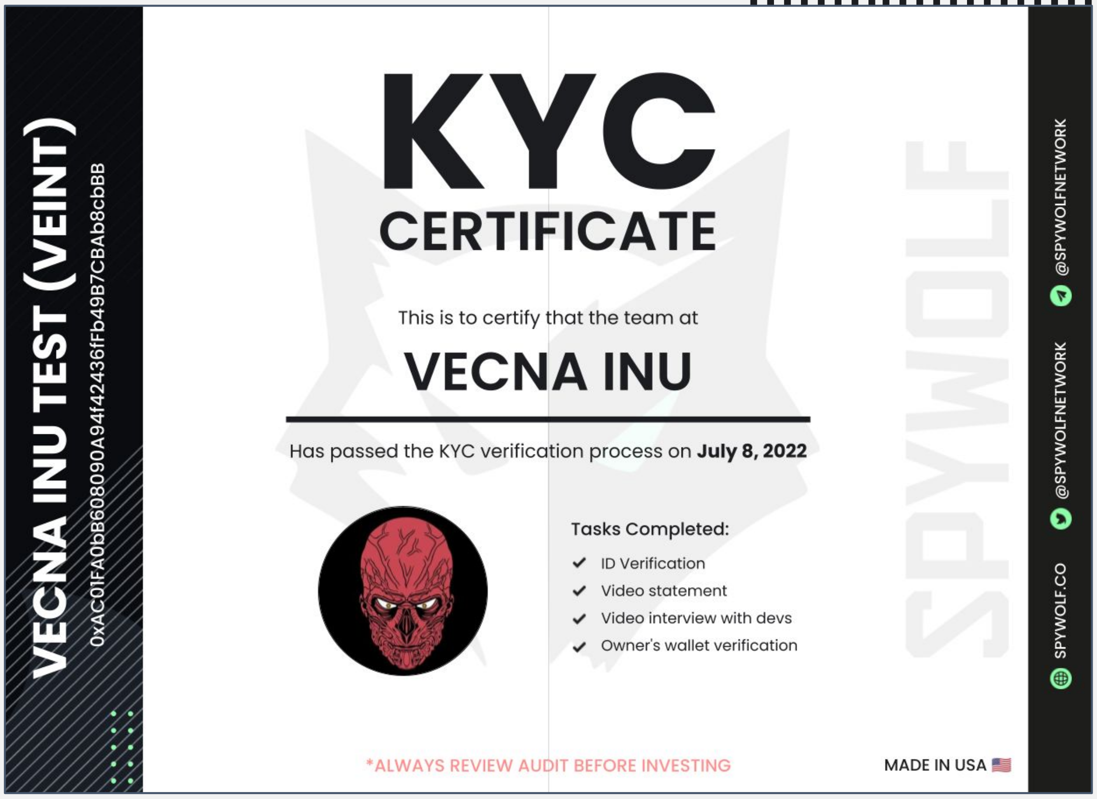
July 8, 2022

Format

Image

Certificate Link

https://github.com/SpyWolfNetwork/KYCs/blob/main/june/KYC_Vecna_INU_0xAC01FA0bB608090A94f42436fFb49B7CBAb8cbBB.png



VECNA INU TEST (VEINT)

0xAC01FA0bB608090A94f42436fFb49B7CBAb8cbBB

KYC CERTIFICATE

This is to certify that the team at

VECNA INU

Has passed the KYC verification process on **July 8, 2022**

Tasks Completed:

- ✓ ID Verification
- ✓ Video statement
- ✓ Video interview with devs
- ✓ Owner's wallet verification

*ALWAYS REVIEW AUDIT BEFORE INVESTING

MADE IN USA 🇺🇸

SPYWOLF

@SPYWOLFNETWORK

@SPYWOLFNETWORK

SPYWOLF.CO

WEBSITE

Website URL
https://vecnainu.com/

Domain Registry
www.publicdomainregistry.com

Domain Expiration
Expires on 2023-06-17

Technical SEO Test
Passed

Security Test
Passed. SSL certificate present

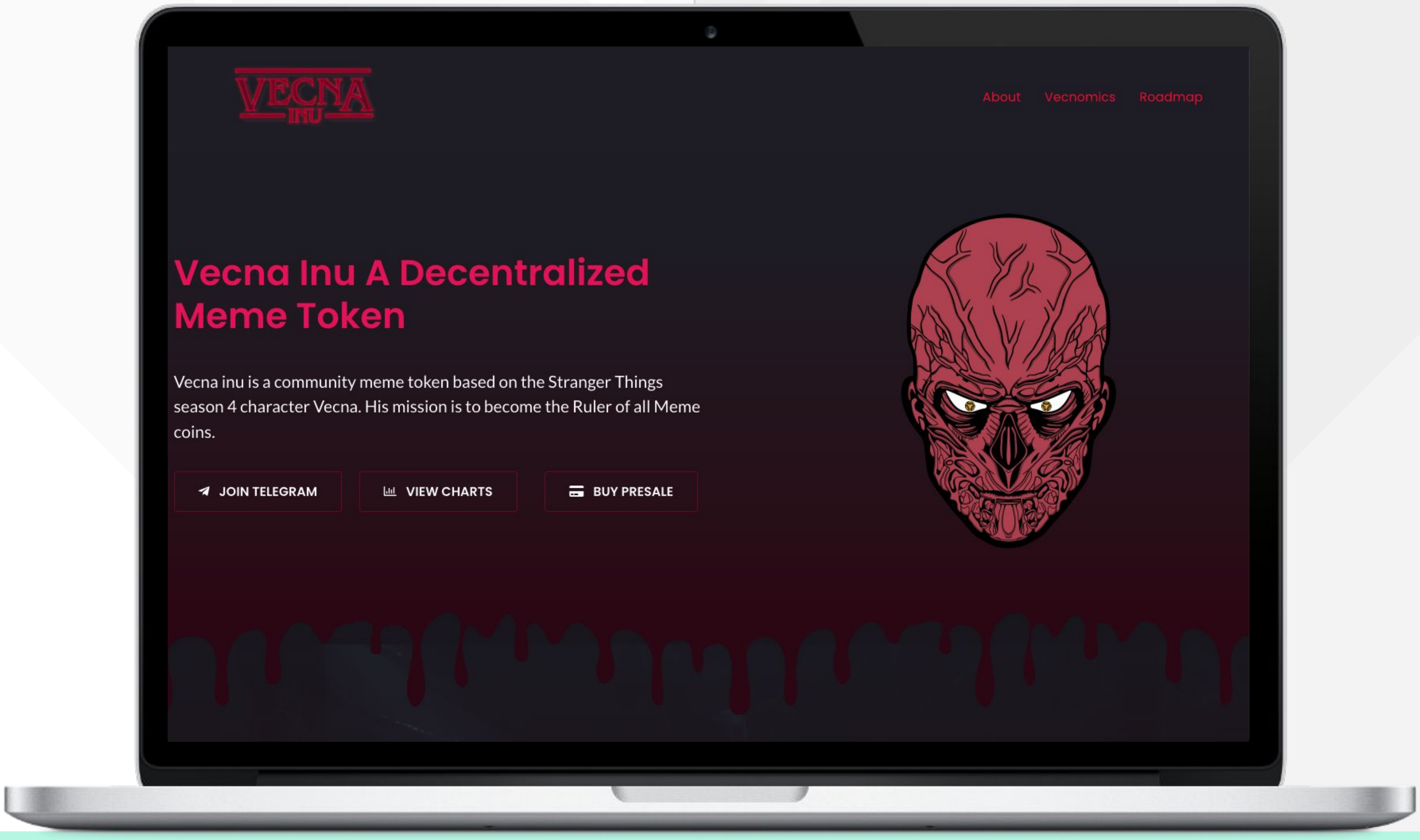
Design
Single page template design, appropriate color scheme.

Content
The information helps new investors understand what the product does right away. No grammar errors found.

Whitepaper
⚠️ No

Roadmap
Yes, goals set at 3 phases without time frames, not many utilities planned.

Mobile-friendly?
Yes



Vecna Inu A Decentralized Meme Token

Vecna inu is a community meme token based on the Stranger Things season 4 character Vecna. His mission is to become the Ruler of all Meme coins.

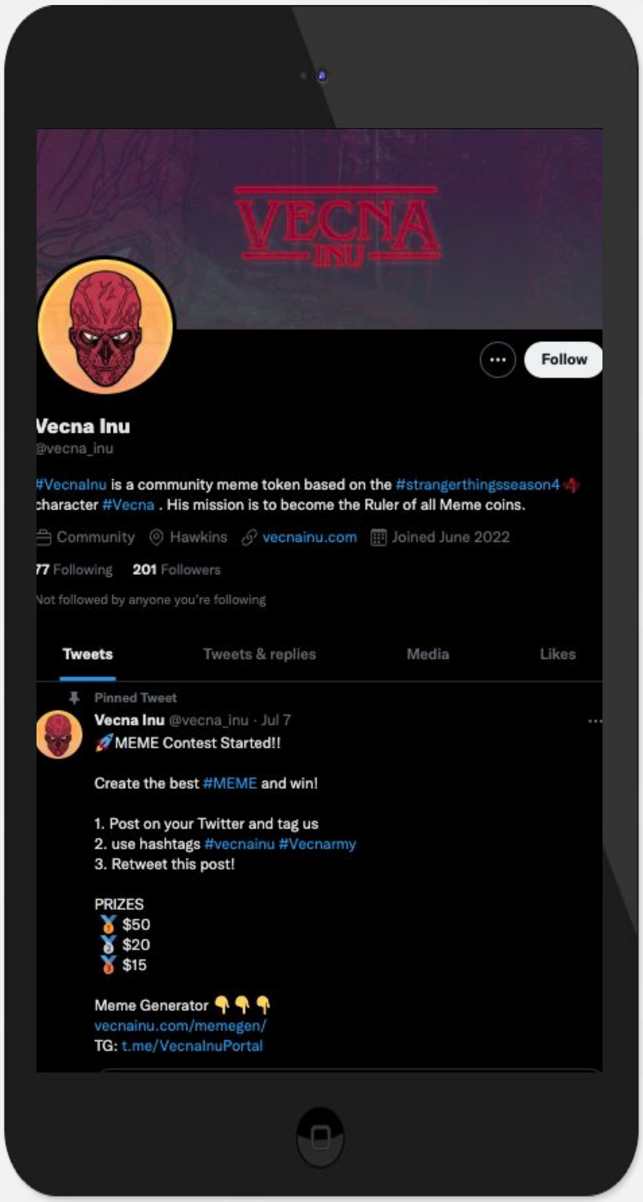
vecnainu.com

SOCIAL MEDIA

& ONLINE PRESENCE

ANALYSIS

Project's social media accounts are active with organic users.



Vecna Inu
@vecna_inu

#VecnaInu is a community meme token based on the #strangerthingsseason4 character #Vecna . His mission is to become the Ruler of all Meme coins.

Community Hawkins vecnainu.com Joined June 2022

77 Following 201 Followers

Not followed by anyone you're following

Pinned Tweet

Vecna Inu @vecna_inu · Jul 7

🚀MEME Contest Started!!

Create the best #MEME and win!

1. Post on your Twitter and tag us
2. use hashtags #vecnainu #Vecnarmy
3. Retweet this post!

PRIZES
🥇 $50
🥈 $20
🥉 $15

Meme Generator 👇👇👇
vecnainu.com/memegen/
TG: t.me/VecnaInuPortal

Twitter

https://twitter.com/vecna_inu

- 207 followers
- Active
- Few posts per day

Discord

- Not available

Telegram

https://t.me/VecnaInuPortal

- 786 members
- Active users
- Active mods

Medium

- Not available

SPYWOLF

CRYPTO SECURITY

Audits | KYCs | dApps
Contract Development

ABOUT US

We are a growing crypto security agency offering audits, KYCs and consulting services for some of the top names in the crypto industry.

- ✓ OVER 150 SUCCESSFUL CLIENTS
- ✓ MORE THAN 500 SCAMS EXPOSED
- ✓ MILLIONS SAVED IN POTENTIAL FRAUD
- ✓ PARTNERSHIPS WITH TOP LAUNCHPADS, INFLUENCERS AND CRYPTO PROJECTS
- ✓ CONSTANTLY BUILDING TOOLS TO HELP INVESTORS DO BETTER RESEARCH

To hire us, reach out to
contact@spywolf.co or
t.me/joe_SpyWolf

FIND US ONLINE

SPYWOLF.CO

SPYWOLF.NETWORK

@SPYWOLFNETWORK

@SPYWOLFOFFICIAL

@SPYWOLFNETWORK

@SPYWOLFNETWORK

Disclaimer

This report shows findings based on our limited project analysis, following good industry practice from the date of this report, in relation to cybersecurity vulnerabilities and issues in the framework and algorithms based on smart contracts, overall social media and website presence and team transparency details of which are set out in this report. In order to get a full view of our analysis, it is crucial for you to read the full report.

While we have done our best in conducting our analysis and producing this report, it is important to note that you should not rely on this report and cannot claim against us on the basis of what it says or doesn't say, or how we produced it, and it is important for you to conduct your own independent investigations before making any decisions. We go into more detail on this in the disclaimer below – please make sure to read it in full.

DISCLAIMER:

By reading this report or any part of it, you agree to the terms of this disclaimer. If you do not agree to the terms, then please immediately cease reading this report, and delete and destroy any and all copies of this report downloaded and/or printed by you. This report is provided for information purposes only and on a non-reliance basis, and does not constitute investment advice.

No one shall have any right to rely on the report or its contents, and SpyWolf and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers and other representatives) (SpyWolf) owe no duty of care towards you or any other person, nor does SpyWolf make any warranty or representation to any person on the accuracy or completeness of the report.

The report is provided "as is", without any conditions, warranties or other terms of any kind except as set out in this disclaimer, and SpyWolf hereby excludes all representations, warranties, conditions and other terms (including, without limitation, the warranties implied by law of satisfactory quality, fitness for purpose and the use of reasonable care and skill) which, but for this clause, might have effect in relation to the report. Except and only to the extent that it is prohibited by law, SpyWolf hereby excludes all liability and responsibility, and neither you nor any other person shall have any claim against SpyWolf, for any amount or kind of loss or damage that may result to you or any other person (including without limitation, any direct, indirect, special, punitive, consequential or pure economic loss or damages, or any loss of income, profits, goodwill, data, contracts, use of money, or business interruption, and whether in delict, tort (including without limitation negligence), contract, breach of statutory duty, misrepresentation (whether innocent or negligent) or otherwise under any claim of any nature whatsoever in any jurisdiction) in any way arising from or connected with this report and the use, inability to use or the results of use of this report, and any reliance on this report. The analysis of the security is purely based on the smart contracts, website, social media and team.

No applications were reviewed for security. No product code has been reviewed.